Preliminary Report on Fake Review Detection

by Andrew Nedilko

Part 1. What is suspicious about the fake reviews? What do you realize about the language used by the reviewers?

To answer these two questions, I examined a number of reviews from this and this files that were rated by Fakespot as real (true) and fake. I explain how I received these two files later in this report. After analyzing the reviews, I have come to the following conclusions.

1.1. Real (True) Reviews

They are more like a story, a narrative. A typical review with a high probability of being true can, for example, describe that someone was in search of a good product, they tried several things, and they finally found this particular product. It's a good product for the money, it has the following features (a list of features follows). It has the following pros and cons.

Such reviews sound natural and sincere. People may speak about their specific unique needs and use cases without utilizing too much of generalized vocabulary. Such reviews may not have a lot of exclamatory words, but even if they do, these words are naturally interwoven into the overall measured, unhurried fabric of the rest of the review.

These are several real examples of reviews about Bluetooth headphones from the Amazon website:

- "Great budget true wireless Earbuds… I have been looking for a good pair of true wireless earbuds for a few months now, and have tried the most expensive ones available, as well as a lot of budget options. When comparing this to others…"
- "Great sound quality, noise cancelling, and comfortable wireless headphones! Verified Purchase I've been using my wireless earbuds for the past couple of days and love them! Comfort/Fit: The earbuds are very cozy and sometimes I forget they are in. It took a couple of times testing out the different sizes of earbud caps to find the perfect fit. They provide you with many different cap sizes which is great…"
- "These are by far the best… There's no noise cancellation option, unless I've missed it. They work fine w Siri. There's no magnetic charging option, but with the long battery life of the charging case, it's hardly necessary… Bottom line--for about 1/6 of the now widely available discounted price of the AirPods Plus, I would seriously consider these."
- "So far, so good.... I received these 2 days ago and I am very impressed. I'll be honest, I thought at the price point that these may not be what was advertised. … I used them yesterday when I cut the grass and the noise cancelling was great, the clarity of sound through these is excellent. … I haven't ridden wearing them yet, so I cannot give a clear indication of the noise cancelling ability and I cannot attest to the battery life either as I have only just used them. All I will say is, so far so good."
- "Good for people who easily lose stuff because they are cheap and you won't feel so bad… Whatever, they sound pretty good. In my household between my husband and I, we have 3 sets

of bose over the ear Bluetooth headphones… Not going to say these are of the same quality or sound but for the sale price of less than $15, and a normal price of $25, I'd buy them again if I lose one, and that is the perfect use for these. Activities where it is possible to lose one."

I have manually composed a small dataset of Amazon reviews and trained ML models on it (see explanation later). You can find more of such real review in this csv file: open it in Excel, sort the 'pred_avg' column in the descending order (preferably in the test set – filter the 'subset' column) – the higher the score, the more real a review is.

Some of the n-grams with very high feature importance scores that I was able to extract from my probabilistic Naïve Bayes model for real reviews included: *'feels', 'lights', 'travel', 'make sure', 'fairly', 'warranty', 'nicely', 'quiet', 'recommended', 'headphones verified', 'complaint', 'hold charge', 'overall great', 'liked', 'annoying', 'ok', 'honestly', 'customer service'*. As you can see, they are really calm, reasonable terms.

1.2. Fake Reviews

In contrast, fake reviews may have a lot of exclamatory words. It may look like an exclamation mark is needed almost after every sentence. Sometimes they get so positive, that the positivity feels almost toxic.

In additional to being overly positive, such reviews don't usually mention anything doubtful about the product (like cons), and if they compare the product with other products, then it's always in favor of the product being reviewed. Any bad or doubtful qualities are left out of fake reviews.

Therefore, this excessive positivity, avoidance of true comparisons, and the overly use of exclamatory language make the fake reviews really suspicious. They may sound too good to be true. Here are some real examples from the same dataset:

- "Lots of Value at a Low Price. I recently purchased the Boloxa A17 wireless headphones, and I must say, they left me **thoroughly impressed**. These headphones are **nothing short of phenomenal**, offering a **fantastic audio experience** coupled with **impressive features** that are hard to believe at **such an affordable price** point…"
- "Headphones, and I must say, they have **exceeded my expectations in every way**! … These earphones have been a **game-changer** for my workout routine… **The ear hooks are fantastic**!..."
- "Bike Riders **Take Note**!! … I need sports headphones that are **TOUGH**, that have **enough battery for a 100 KM ride**, that stay in my ears over the most challenging terrain, that provide **high quality sound** so that whatever I play, **I can hear it clearly**."
- "**Great** wireless headphones! These are my first wireless headphones. **What a great product** at **such a reasonable price**. The sound quality is **amazing**. I am not very tech savvy but found it **easy to pair** with my Android phone. I would **definitely recommend** these headphones to others."
- "**GREAT EARPHONES**. **Great bang for the buck**!! **Do yourself a favor and buy these earphones**!! They are **definitely good quality** and worth the price…"

You can find more of such fake reviews in this csv file: open it in Excel, sort the 'pred_avg' column in the ascending order (preferably in the test set – filter the 'subset' column) – the lower the score, the more fake a review is.

Some of the n-grams with very high feature importance scores that I was able to extract from my probabilistic Naïve Bayes model for fake reviews included: *'really like', 'phenomenal', 'life excellent', 'affordable price', 'unbeatable', 'exceeded expectations', 'crystal clear', 'vigorous', 'definitely recommend', 'particularly', 'high quality', 'outstanding', 'exactly', 'truly', 'exceptional'*. As you can see, they sound overly positive, compelling, exclamatory.

Part 2. What kind of model you would build in order to find the suspicious patterns. What feature engineering you would conduct and the model architecture that would be ideal for this use case.

Baseline Classifier

To assist me in answering the questions in this report, besides just eye-balling the reviews on the Amazon website, I decided to build a baseline machine learning (ML) model. The results and the code are available in this repository.

To collect a dataset, I manually copied and pasted several hundred reviews from the Amazon website into this file for fake reviews and this file for true reviews. I used the Fakepost F and A ratings for this. I then used this and this Python code to parse the data and convert it into a table format. Besides text, I was able to extract such additional features as title, reviewer name, date, # of stars, product color, whether the reviewer is a vine voice (highly ranked reviewer), whether the review has images, how many people found the review useful, whether this was a verified purchase. An EDA was done on this data in this Jupyter notebook.

This Jupyter notebook will tell you a story how the final full joint dataset was deduplicated (to avoid data leakage), the text was cleaned, the stratified train test split was performed in preparation for the classification effort. The dataset contains about ~400 reviews in each of the two categories: fake and real reviews.

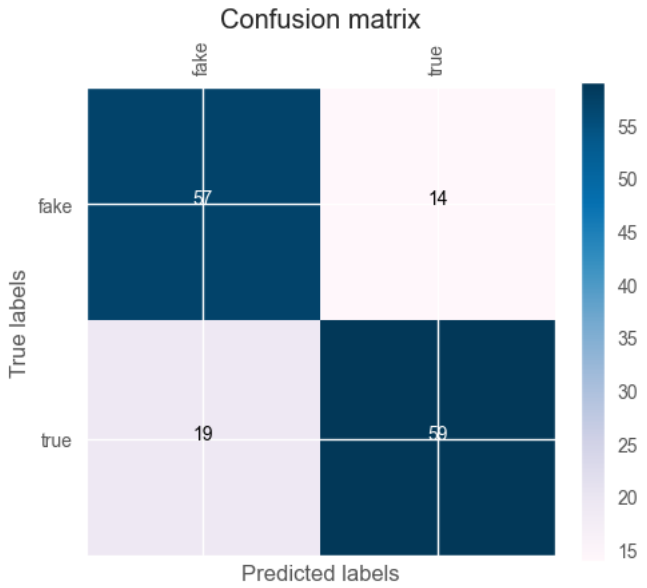
The notebook also contains the code for training an initial binary Logistic Regression model as my baseline classifier. I also trained an initial Naïve Bayes model because this is a probabilistic model, and it's possible to extract the feature importance for each ngram at the model level (not just for one predicted text). This is a useful quality which I coded up in the same notebook. I used it to try to understand what kind of vocabulary is important for the model to classify each class correctly.

In addition, I also trained a more advance initial version of the XGBoost classifier. The features for all the three models were sparse vectors representing text as bag-of-words ngrams based on TF-IDF scores or counts extracted from text by TfidfVectorizer or CountVectorizer. NOTE THAT DUE TO COMPLETE LACK OF TIME, all these models were trained as initial models **without cross-validation or any significant hyperparameter gridsearch,** but the models don't seem to be overfit. Nevertheless, I was able to apply these models back on the dataset to try to understand which reviews are most fake ones and which reviews are most real ones (this code is also in the same notebook), and I received very meaningful results provided in this csv file (if you sort it by the 'pred_avg' column – the lower the number, the more fake a review is). Therefore, my attempt to reverse engineer the Fakespot reviews classifier was in a way successful.

The initial models were quite successful in classifying fake reviews on this small dataset. The Logistic Regression achieved an accuracy / macro- and micro-F1 scores of 0.77 with the precision for the fake class being slightly less than for the true class:

	precision	recall	f1-score	support
fake	0.75	0.79	0.77	71
true	0.80	0.76	0.78	78
accuracy			0.77	149
macro avg	0.77	0.77	0.77	149
weighted avg	0.77	0.77	0.77	149

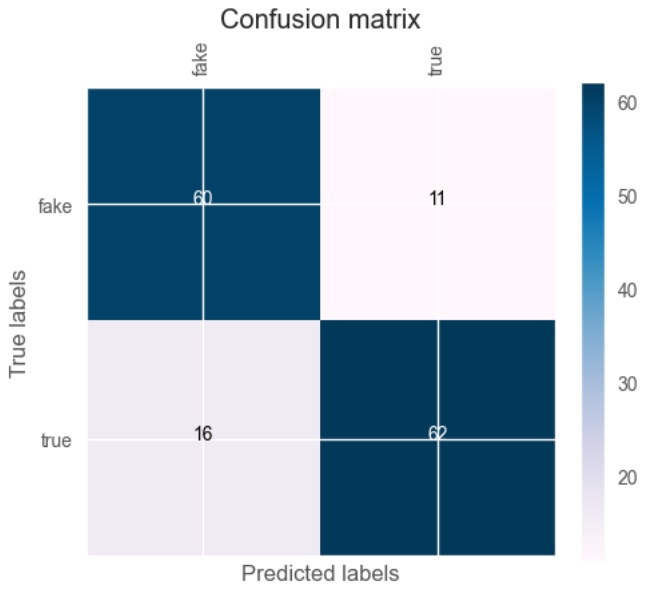
Confusion matrix:



The XGBoost model had even better results which was expected – an accuracy / macro- and micro-F1 scores of 0.82:

	precision	recall	f1-score	support
fake	0.79	0.85	0.82	71
true	0.85	0.79	0.82	78
accuracy			0.82	149
macro avg	0.82	0.82	0.82	149
weighted avg	0.82	0.82	0.82	149

Confusion matrix:



One way to improve the BoW models can be using additional non-text features (see below). I could have taken the probability predicted by the text classifier and combined it with other numerical features in another model. Or I could have trained an ensemble model for text + non-text features. An ensemble model can also improve the results if it consists just of several text classifiers (and then combined with numeric features).

Subsequent ML Experimentation

It was my intent to conduct these experiments too, but one and a half days were drastically not enough time to accomplish such a task. That is why I will just describe what else can be attempted to build a reliable fake review classification system.

In addition to the above BoW models, more advanced model architectures and features may include:

- A deep learning convolution neural network (CNN) which proved to be useful for text classification as well for image classification.
- A sequence model – as simple as one or several dense fully connected layers or as complex as a bidirectional LSTM model (long short-term memory) or several GRU layers (gated recurrent unit). A hyperparameter search can help determine the best architecture for the CNN and RNN networks including the number and type of layers, the number of units per layer, the activation functions, the loss function, the type of optimizer, the number of training epochs, when to implement early stopping to avoid overfitting (e.g. after reaching a certain accuracy, loss, F1 score), other training objectives and strategies.

  The input for CNNs and RNNs can be vectorized text in the form of numbers where each word is assigned a specific number in the vocabulary – the sequence of words is represented as a sequence of numbers and sent to the model's input layer which is followed by an embedding layer which, in turn, learns the new word embeddings from scratch for our texts (dense vectors for each word). We can also use the sparse vectors as described at the precious step (TF-IDF scores or counts).

  The third option would be to use pretrained word embeddings such as word2vec, GLOVE, contextual word embeddings (from the ELMo, ULM-Fit, BERT, Sentence-BERT, GPT and other transformer models). In this case, we don't need to learn embeddings from scratch. Another notable element of the model architecture can be the second input layer for non-text features

(see below) which can be learned as embeddings too and then concatenated with the text embeddings to form joint input that will pass through the remaining layers in the model.

- Another option would be to fine-tune an encoder transformer model like BERT base, RoBERTa, DeBERTa, or DistilBERT. The last option will probably be the fastest, but all these models will have a relatively high latency (long model response). To mitigate this, we can use knowledge distillation when a transformer model is used to make prediction on the dataset, and this dataset along with the prediction (logits) is used to train a simpler student model. We can also try quantization as provided by some frameworks like PyTorch, for example. I used this approach successfully for a very difficult task of implicit bias classification in text.
- The last option would be to use a decoder transformer model like GPT-3, ChatGPT or GPT-4. The zero-shot method will probably not work because of the complexity of the task. The few-shot method will hardly work too because it allows showing only a limited number of examples to the model because the model has a limited context window size. The best approach here would be to fine-tune either a GPT-3 model like DaVinci or even fine-tune ChatGPT which can be done starting from last August, I believe. A fine-tuned GPT model can have exceptional performance as proved by my experience at the recent ACL conference where I published two papers about amotion detection. My fine-tuned model GPT-3 scored the second among all the participants of a shared task for emotion detection at this conference.
- Trade-off: when a model is productionized, it is always a tradeoff between the model accuracy and latency. Accurate model like BERT or GPT have billions of model parameters and have a high latency. If we don't want the user to wait for model response for too long, we can deploy lighter models like the above BoW or even RNN models. But they may have lower accuracy. Although from my experience, a sequence model like RNN can be comparable with BERT for some of the tasks in terms of accuracy.

Candidates for Additional Features

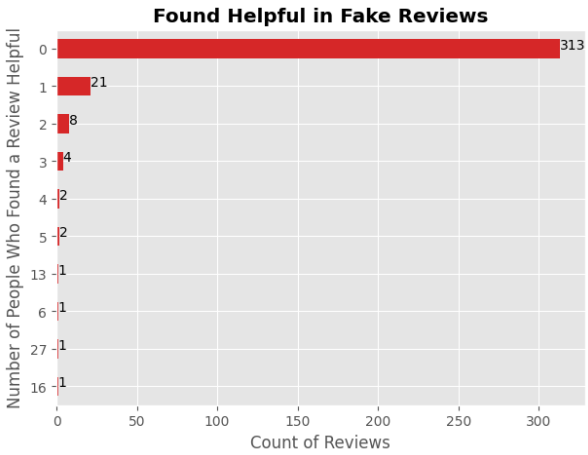
Based on the EDA results on my dataset, I can say that the following non-text based features may have some potential predictive power for fake review detection. Additional research is needed to confirm this hypothesis.

- For each product: **the distribution of the number of reviews for each number of stars**. Fake reviews tend to have most of their reviews in the five-star category, leaving very few reviews in the one-, two-, three-, and four-star categories. The real reviews may have a more realistic distribution by the number of stars and may have a relatively large number of reviews in other than five-star categories.

- Encoding information from the plots depicting the number of reviews vs. product price shown in the Fakespot analysis reports may provide value in certain cases. On an example plot below, you can see that while the number of very positive reviews grew over time, for some reason the product price continued to fall which didn't make sense. This can be a sign that the product was not selling as expected, and the company tried to bump up sales by simultaneously writing

fake reviews and dropping the price.

- True reviews have a greater **number and variety of colors**. 270 out of 390 true reviews where the product color was indicated (vs. 114 out of 354 for fake reviews) and the colors in real reviews include: white, black, green, navy blue, dark green, shadow gray, purple, rose gold, earhooks-gold (vs. only white, black, and 1 red for fake reviews).
- **Found helpful** – real reviews are found helpful considerably more often (by count) and by a lot more people (per review and total) as opposed to fake reviews.



- There are more **verified purchases** among true reviews (363) vs. fake reviews (287), but their number is still high for fake reviews.
- **Reviewer name** – there only 4 people with repeated names in the true reviews, and 10 such people in the fake reviews. One would expect this number to be higher for fake reviews [1], but maybe the fake reviewers have become smarter and use different names or, which is most probably the case here, the size of the dataset is still insufficient to make a final conclusion on this matter.
- Surprisingly, there are more so-called **vine voices** who wrote fake reviews than true reviews, but this may be due to the small size of the dataset too. Vine voices are special, highly ranked reviewers on Amazon.com.
- Fake reviewers also seem to post more **images** along with their reviews.

Potential Concern

With the rise of Generative AI, the fake review detection task may gradually become closer to the AI generated text detection task. It has become very easy to use ChatGPT and other models for composing fake reviews in an attempt to disguise them or to overcome the writer's block which fake reviewers may experience, especially when writing a lot of reviews. The thing is that you can ask an LLM to rewrite the same review using different styles without repeating the same words!

Literature Review

Provided I had more time, I would have written a literature review based on the references listed below to see what kind methods are used nowadays for fake review detection. It seems like there are many publications on this topic, and some of them are quite recent.

References

1. Domenico Delle Side. Rome wasn't built in a day: spotting fake reviews: https://www.kaggle.com/code/nicodds/rome-wasn-t-built-in-a-day-spotting-fake-reviews
2. Daniel Martens, Walid Maalej. Towards Understanding and Detecting Fake Reviews in App Stores. 2019. https://arxiv.org/pdf/1904.12607.pdf
3. Joni Salminen, Chandrashekhar Kandpal, Ahmed Mohamed Kamel, Soon-gyo Jung, Bernard J. Jansen. Creating and detecting fake reviews of online products. 2022. https://www.sciencedirect.com/science/article/pii/S0969698921003374
4. Aaryan Rustagi1, Vajraang Padisetti1, and Suresh Subramaniam. Fake Review Detection Using Machine Learning. 2022. https://www.jsr.org/hs/index.php/path/article/view/3281/1215
5. Begüm Yılmaz. Fake Review Detection in 2023: Overview, Methods & Case Studies. 2023. https://research.aimultiple.com/fake-review-detection/
6. René Theuerkauf, Ralf Peters. Detecting Fake Reviews: Just a Matter of Data. 2023. https://scholarspace.manoa.hawaii.edu/server/api/core/bitstreams/b9d8097d-81d5-478b-9d77-8b1397f43aeb/content
7. Deception Detection in Amazon Reviews. https://github.com/aayush210789/Deception-Detection-on-Amazon-reviews-dataset
8. Plotkina, D., Munzel, A., & Pallud, J. (2020). "Illusions of truth—Experimental insights into human and algorithmic detections of fake online reviews." *Journal of Business Research*, 109, 511-523. Retrieved January 17, 2023.
9. "Total number of user reviews and opinions on Tripadvisor worldwide from 2014 to 2021". *Statista*. February 21, 2022. Retrieved January 17, 2023.
10. Li, J., Lv, P., Xiao, W., Yang, L., & Zhang, P. (2021). "Exploring groups of opinion spam using sentiment analysis guided by nominated topics." *Expert Systems with Applications*, *171*, 114585. Retrieved January 17, 2023.

11. Saumya, S., & Singh, J. P. (2018). "Detection of spam reviews: a sentiment analysis approach." *CSI Transactions on ICT*, *6*(2), 137-148. Retrieved January 17, 2023.

12. Jain, P. K., Pamula, R., & Ansari, S. (2021). "A supervised machine learning approach for the credibility assessment of user-generated content." *Wireless Personal Communications*, *118*(4), 2469-2485. Retrieved January 17, 2023.

13. "Amazon targets fake review fraudsters on social media." *Amazon*. July 19, 2022. Retrieved January 17, 2023.

14. Detecting Fake Job Postings Using Python and Keras. https://www.kaggle.com/code/madz2000/text-classification-using-keras-nb-97-accuracy

15. Detecting Fake Job Postings Using R. https://rpubs.com/abbylmm/fake_job_posting